

LIST OF CLAIMS / AMENDMENTS

In the Claims

Please amend claims 1-14, 28, and 30 as shown herein.

Please cancel claim 29 without prejudice.

Claims 15-27 and 34-45 were previously canceled.

Claims 1-14, 28 and 30-33 are pending and are listed following:

1. (currently amended) ~~An~~ A testing interface device for testing an in-test host's support of USB peripherals, the testing interface device comprising:

~~one or more~~ USB interfaces configured to communicate with one or more USB ports of the in-test host to communicate USB messages with the in-test host;

a network interface configured to communicate with a peripheral emulator using a network communications protocol which supports logical network ports, the peripheral emulator configured to emulate USB peripheral devices;

operating logic configured to perform actions comprising:

receiving USB command messages sent from the in-test host to the testing interface device;

sending the USB command messages from the testing interface device to the peripheral emulator through the network interface using the network communications protocol; and

1 receiving USB response messages sent from the
2 peripheral emulator to the testing interface device through the
3 network interface using the network communications
4 protocol;

5 sending the received USB response messages from the
6 testing interface device through the ~~one or more~~ USB
7 interfaces to the in-test host to determine whether the in-test
8 host supports proper operation of the emulated USB devices;
9 and

10 maintaining a correspondence between the emulated
11 USB peripheral devices and the logical network ports of the
12 testing interface device, such that upon receiving a USB
13 command message from the in-test host using a USB protocol
14 and corresponding to a particular emulated USB peripheral
15 device, the testing interface device sends the USB command
16 message to the peripheral emulator via one of the logical
17 network ports which corresponds to the particular emulated
18 USB device, and such that when receiving a USB response
19 message from the peripheral emulator using the network
20 communications protocol and corresponding to the particular
21 emulated USB peripheral device, the testing interface device
22 receives the USB response message via the logical network
23 port which corresponds to the particular emulated USB
24 device.
25

1
2 **2. (currently amended)** ~~An~~ A testing interface device as recited
3 in claim 1, further comprising the peripheral emulator, wherein the peripheral
4 emulator is programmed to emulate one or more USB peripherals.

5
6 **3. (currently amended)** ~~An~~ A testing interface device as recited
7 in claim 1, further comprising the peripheral emulator, wherein the peripheral
8 emulator is programmed to emulate HID, bulk, and isochronous USB peripherals.

9
10 **4. (currently amended)** ~~An~~ A testing interface device as recited
11 in claim 1, further comprising the peripheral emulator, wherein the peripheral
12 emulator comprises a general-purpose computer programmed to emulate one or
13 more USB peripherals.

14
15 **5. (currently amended)** ~~An~~ A testing interface device as recited
16 in claim 1, further comprising the peripheral emulator, wherein the peripheral
17 emulator comprises a general-purpose computer programmed to emulate HID,
18 bulk, and isochronous USB peripherals.

1 6. (currently amended) An A testing interface device as recited
2 in claim 1, further comprising the peripheral emulator, wherein:
3 the peripheral emulator comprises a general-purpose computer;
4 the general-purpose computer is programmed to emulate one or more USB
5 peripherals; and
6 the general-purpose computer is further programmed to generate USB
7 response messages that test the in-test host with ranges of USB peripheral
8 parameters.

9
10 7. (currently amended) An A testing interface device as recited
11 in claim 1, further comprising the peripheral emulator, wherein:
12 the peripheral emulator comprises a general-purpose computer;
13 the general-purpose computer is programmed to emulate one or more USB
14 peripherals; and
15 the general-purpose computer is further programmed to generate abnormal
16 USB response messages in order to test the in-test host with such abnormal USB
17 response messages.

1 8. (currently amended) An A testing interface device as recited
2 in claim 1, wherein:

3 ~~a particular USB command message is designated for a particular one of a~~
4 ~~plurality of different emulated peripheral devices;~~

5 ~~the network communications protocol supports a plurality of logical ports;~~

6 ~~the operating logic maintains a correspondence between emulated~~
7 ~~peripheral devices and logical ports; and~~

8 the operating logic is further configured to automatically send
9 acknowledgment messages from the testing interface device to the in-test host
10 while waiting to receive the USB response messages from the peripheral emulator
11 sends said particular USB command message to one of the logical ports that
12 corresponds to said particular one of the plurality of different emulated peripheral
13 devices.

14
15 9. (currently amended) An A testing interface device as recited
16 in claim 1, wherein the one or more USB interfaces comprise at least four USB
17 interfaces.

18
19 10. (currently amended) An A testing interface device as recited
20 in claim 1, wherein the USB messages comprise HID, bulk, and isochronous USB
21 messages.
22
23
24
25

11. (currently amended) ~~An~~ A testing interface device as recited
in claim 1, wherein the network interface comprises an Ethernet interface.

12. (currently amended) ~~An~~ A testing interface device as recited
in claim 1, wherein the network communications protocol comprises an Ethernet
communications protocol.

13. (currently amended) ~~An~~ A testing interface device as recited
in claim 1, wherein the network communications protocol comprises an IP
protocol.

14. (currently amended) ~~An~~ A testing interface device as recited
in claim 1, wherein the network communications protocol comprises UDP over IP.

15-27. (canceled)

28. (currently amended) A method of testing an in-test host's support of USB peripherals, comprising:

- sending USB command messages from an in-test host to a testing interface device which supports logical network ports;
- receiving the USB command messages from the in-test host at ~~an~~ the testing interface device;
- packaging the received USB command messages in command data packets formatted in accordance with a network communications protocol;
- sending the command data packets ~~from~~ via the logical network ports of the testing interface device to ~~one or more peripheral emulators~~ a peripheral emulator over network communications media;
- unpackaging the USB command messages from the command data packets at the peripheral emulator;
- emulating one or more USB peripheral devices that respond to the USB command messages which have been unpackaged by generating USB response messages;
- packaging the USB response messages in response data packets at the peripheral emulator, such that the response data packets can be sent from the peripheral emulator to the testing interface device using the network communications protocol;
- receiving the response data packets from the ~~one or more peripheral emulators~~ peripheral emulator over the network communications media at the testing interface device via the logical network ports, wherein the response data packets are formatted in accordance with a network communications protocol;

1 unpackaging USB response messages from the received response data
2 packets at the testing interface device;
3 sending the ~~unpackaged~~, USB response messages which have been
4 unpackaged from the testing interface device to the in-test host;
5 receiving the USB response messages at the in-test host;
6 determining whether the in-test host supports proper operation of the
7 emulated USB devices based on the USB response messages; and
8 maintaining a correspondence between the emulated USB peripheral
9 devices and the logical network ports of the testing interface device, such that
10 upon receiving a USB command message from the in-test host using a USB
11 protocol and corresponding to a particular emulated USB peripheral device, the
12 testing interface device sends the USB command message to the peripheral
13 emulator via one of the logical network ports which corresponds to the particular
14 emulated USB device, and such that when receiving a USB response message
15 from the peripheral emulator using the network communications protocol and
16 corresponding to the particular emulated USB peripheral device, the testing
17 interface device receives the USB response message via the logical network port
18 which corresponds to the particular emulated USB device.

19 **29. (canceled)**
20
21
22
23
24
25

1 **30. (currently amended)** A method as recited in claim 28, further
2 comprising ~~creating abnormal USB response messages in response to the~~
3 ~~packaged USB command messages and packaging said abnormal USB response~~
4 ~~messages in the response data packets~~ emulating one or more abnormal USB
5 peripherals that respond to the USB command messages which have been
6 unpackaged, by generating abnormal USB response messages and packaging the
7 abnormal USB response messages in response data packets in order to test the in-
8 test host's ability to handle such abnormal USB response messages.

9
10 **31. (original)** A method as recited in claim 28, wherein the network
11 communications protocol comprises an Ethernet communications protocol.

12
13 **32. (original)** A method as recited in claim 28, wherein the network
14 communications protocol comprises an IP protocol.

15
16 **33. (original)** A method as recited in claim 28, wherein the network
17 communications protocol comprises UDP over IP.

18
19 **34-44. (canceled)**
20
21
22
23
24
25